**Student name: \_\_\_\_\_Charlotte Godley\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Semester: \_\_\_1\_\_ Week: \_12\_\_ Meeting number: \_\_\_7\_\_**

**Pre-meeting:**

|  |
| --- |
| **Achievements (relate to last week’s aims):**   * Built a wrapper class for the SAX API, which takes a folder and list of tags to search for, iterates through each xml or mxml file in the given folder and extracts what's in it to a dictionary, indexed by tag type, and then what the tag contains. This is then serialized to a file named “.metadata” in the given folder * drew out a mindmap of elements in music and how they link together. From this created a basic class diagram: intending on storing things by measure, containing musical settings (timesigs and where they appear in time, same for clefs and metronome marks), a list of parts containing dynamic/phrase/style/slur dicts, notes and the name of the instrument. * Found a python library encompassing a lot of the same features – [music21](http://web.mit.edu/music21/) – designed for musicology analysis but provides a good base for objects as some things, such as beaming of notes, I'd missed from my class diagram.   + Considering this is developed by MIT and they choose to use lilypond instead of developing their own rendering system, thinking it would be easiest to generate lilypond formatting code and display music as a pdf inside a pyQt window? Would this be allowed? <http://c2.com/cgi/wiki?LilyPond> : want to avoid reinventing the wheel. Lilypond and MusiTex are more of formatting languages than software packages, both are OpenSource. Need to check license.     - PDF may also solve issues of panning and zooming, depending on python GUI integration. Need to look at this in the next week * Looked briefly at how music21 generates MIDI: think this is doable but need to do more on this this week to confirm. Had problems with ACW for other module or else would have worked more on this |
| **Reflection:** |

**At meeting:**

|  |
| --- |
| **Aims and objectives:**  - review lilypond's links in other programs to see how interaction is built with regard to error messages and look at syntax of lilypond to gain familiarity. Also look at PDF and PNG viewers in python as if lilypond is included, graphics will be rendered in a viewing window.  - review MIDI and how you interact with it from a programming point of view to see whether it's potentially viable as a feature inclusion |
| **Other comments:** |